Improved Uncalibrated View Synthesis by Extended Positioning of Virtual Cameras and Image Quality Optimization

Fabian Gigengack and Xiaoyi Jiang

Department of Mathematics and Computer Science, University of Münster,
Einsteinstraße 62, 48149 Münster, Germany
{fabian.gigengack,xjiang}@uni-muenster.de
http://cvpr.uni-muenster.de

Abstract. Although there exist numerous view synthesis procedures, they are all restricted to certain special cases. Some procedures for instance can only handle a calibrated camera set while others are limited to interpolation between the reference views. In this paper we will present a fully automated uncalibrated view synthesis procedure. It allows an arbitrary camera placement in 3-D space on the basis of only two input images with a natural camera orientation. Natural camera orientation means that the focus of the virtual camera is intrinsically given by the geodesic which again is determined by the reference views. The presented procedure extends an existing view synthesis algorithm that allows only a camera placement on the 1-D geodesic (in the case of two reference views). The extensions are an additional camera placement along and orthogonally to the line of sight. The image quality of the virtual views will also be enhanced by utilizing the image information of both reference views.

Keywords: Uncalibrated View Synthesis, Relative Affine Structure.

1 Introduction

The term *view synthesis* denotes the generation of virtual views of a scene based on a few reference pictures of this scene. This paper concentrates on the case of exact two reference pictures. It will be shown that such two reference views are sufficient for a flexible camera placement in 3-D space by means of an approach for *uncalibrated* view synthesis *without* the need of any *user interaction*.

Motivation. View synthesis techniques can be applied to various fields. In the film industry for instance, it is widely used to create the 'bullet time' effect which became famous with the 1999 movie 'The Matrix'. By using multiple cameras and view synthesis algorithms, a virtual camera movement around a scene which seems to be frozen in time is simulated.

H. Zha, R.-i. Taniguchi, and S. Maybank (Eds.): ACCV 2009, Part II, LNCS 5995, pp. 438–447, 2010.

Another field of application is sports. View synthesis techniques were already used during the *European Soccer Championship 2008*, where important situations were analyzed with a computer program called *LiberoVision*[1]. Decisions about offsides for example can be supported by placing a virtual camera on a level with the affected players.

Another example of use in the range of sports could be goalkeeper practice in handball. A penalty shot can possibly determine whether a game will be won or lost. To prepare the goalkeeper for opposing players, the video material of past penalty shots of these players could be analyzed. A view synthesis algorithm together with cameras next to the goal allow to simulate the point of view of the goalkeeper for a realistic training.

Related Work. The roots of *view synthesis* can be found for example in the work of S. Seitz and C. Dyer [1] from 1996. They present a method called *view morphing* which is a combination of *view interpolation* and *image morphing*. Restrictions of this technique are that some amount of user interaction is necessary to mark feature points and only interpolated views can be generated.

A technique reminding of the already mentioned ‘bullet time’ effect was introduced in 2004 by Zitnick et al. [2]. View interpolation is performed between sparse synchronized cameras arranged along a one dimensional arc. A drawback of this technique is the limitation to interpolation only and the need of a calibrated camera system.

The 2007 paper of Criminisi et al. [3] addresses the issue view synthesis for teleconferencing and provides satisfying results. It generates new views with the constraint of a pair of rectified video streams.

Techniques without the need of a calibrated camera system have been proposed for example by S. Avidan and A. Shashua [4] in 1998. They established the so-called *trilinear tensor* for three reference images which can be used to describe the spatial relationships without prior calibration. The technique also allows an extrapolation of the views but has the general drawback of requiring some user interaction.

An automated procedure was presented in 2007 by A. Fusiello et al. [5,6] with focus on a novel positioning method of virtual cameras. The possible new camera positions for two reference views lie on a curve (1-D manifold) through the two reference cameras, allowing interpolation as well as extrapolation.

Contribution of the Paper. Based on the algorithm in [6], which will be introduced shortly in Section 2, we have developed methods that extend the camera placement from a 1-D manifold to arbitrary positions in 3-D space. These extensions are described in Section 3. Apart from the flexible camera placement the key benefits are the abandonment of a manual camera calibration and any kind of user interaction. Further in Section 4 a method for image quality improvement will be discussed based on the utilization of the image information of both reference images. Some experimental results will be discussed in Section 5 followed by a general conclusion in Section 6.

[1] `http://www.liberovision.com/`

2 Uncalibrated View Synthesis

The view synthesis algorithm as presented by A. Fusiello in [6] has a pipeline structure [5]. This has the general advantage that every single step of the pipeline can be realized by a suitable method. The pipeline consists of the following steps:

1. Parameter estimation *(keypoint detection, outlier elimination, rectification of the input images)*
2. Stereo analysis *(stereo matching, de-rectification of the disparity maps, calculation of the relative affine structure)*
3. Warping of the input images

In this work we will only discuss the third part of the pipeline, the warping of the input images. A detailed disquisition on the other two parts can be found in [5].

An important part of the warping procedure is represented by the so-called *uncalibrated rigid transformation matrices* which control the virtual camera movement between two views as we will see later on.

Definition 1. *We define the* uncalibrated rigid transformation matrix *as a combination of the homography at infinity* $H_{\infty 12}$ *between image* 1 *and image* 2 *and the epipole* e_2 *of the second image*

$$D_{12} := \begin{bmatrix} H_{\infty 12} & e_2 \\ 0 & 1 \end{bmatrix}. \tag{1}$$

As shown in [6], we can identify the uncalibrated rigid transformation matrices with the well understood transformation of the *special euclidean group* $SE(3, \mathbb{R})$. Since $SE(3, \mathbb{R})$ is a Lie group it is possible to compute continuously varying interpolated and extrapolated virtual camera positions by scaling and combining transformation matrices.

The handling of the special euclidean group $SE(3, \mathbb{R})$ is realized through the following definitions from [6]:

Definition 2. *Given* $G \in SE(3, \mathbb{R})$*, the* scalar multiple *of* G *is defined as*

$$G^t := \exp(t \log(G)),\ t \in \mathbb{R} \tag{2}$$

As G in the definition is a 4×4 matrix the function identifier exp and log describe the matrix exponential and matrix logarithm. Matrices of $SE(3, \mathbb{R})$ can also be combined by a *linear combination.*

Definition 3. *Let* G_1*,* $G_2 \in SE(3, \mathbb{R})$*. The* linear combination *of* G_1 *and* G_2 *is defined as*

$$(G_1^u) \oplus (G_2^v) := \exp(u \log(G_1) + v \log(G_2)),\ u, v \in \mathbb{R} \tag{3}$$

Two linear independent uncalibrated rigid transformation matrices span a two dimensional manifold in $SE(3, \mathbb{R})$.

With these tools applied to uncalibrated rigid transformation matrices, a point m_1 from the first image can be transformed to its corresponding point m_2 in the second image with the following formula:

$$m_2 \cong [I|0] D_{12} \begin{pmatrix} m_1 \\ \gamma_1 \end{pmatrix} . \tag{4}$$

The symbol $\cong$ means 'equal up to a scale factor' and I denotes the identity matrix. The term γ_1 represents the *relative affine structure* of m_1. The theory of the relative affine structure was introduced by A. Shashua and N. Navab in [7] and [8]. It is used to describe the depth information of the scene relative to the corresponding reference view and is gained in our case from the disparity maps, which can be computed from the rectified reference views.

In order to perform the coordinate transformation in Equation (4) the relative affine structure γ_1 and the uncalibrated rigid transformation matrix D_{12} are needed. We will now briefly introduce the computation of the matrix D_{12} while further information on the derivation of the relative affine structure can be found in [6]. Referring to Definition 1 we need to compute the homography at infinity $H_{\infty 12}$ and the epipole e_2 of the second image. The homography at infinity can be easily obtained from the precomputed homographies of the first (H_1) and second (H_2) reference image out of the rectification step (see [9]):

$$H_{\infty 12} = H_2^{-1} H_1 . \tag{5}$$

The next step is to compute the epipole e_2. As the fundamental matrix F is known from the parameter estimation step in the pipeline (see [5] for further information) and $e_2^T F = 0$ holds, the epipole e_2 is given as the left zero vector of F.

With this, the matrix D_{12} is fully defined. For $\nu \in \mathbb{R}$ and a point m_1 in the first reference image the new position m_ν of m_1 in the virtual image defined by $D_{1\nu} := D_{12}^{\nu}$ can be calculated according to Equation (4) via

$$m_\nu \cong [I|0] D_{1\nu} \begin{pmatrix} m_1 \\ \gamma_1 \end{pmatrix} . \tag{6}$$

3 Extended Positioning of Virtual Cameras

The method described in Section 2 is already a powerful tool for generating virtual views of a scene. In order to extend the potential of this algorithm we will present some refinements that augment the variety of camera positioning from a 1-D curve (geodesic) to an arbitrary point in 3-D space. In Section 3.1 we introduce a procedure that allows additional to the camera positions on the geodesic a camera movement along the line of sight and in Section 3.2 this will be extended to a camera movement orthogonally to the geodesic and the line of sight.

3.1 Camera Movement along the Line of Sight

For the purpose of a more flexible virtual camera placement we will discuss an extended camera movement along the line of sight[2] in this section.

The principle can be seen in Figure 1. L and R denote the two real cameras on the left and on the right. Between them is a virtual camera V which lies on the geodesic describing an interpolation. The observed scene is indicated by the big ball. The dashed line represents the extended camera positions along the line of sight for V. Remember that V is chosen randomly. The two small balls on the dashed line indicate possible new camera placements.

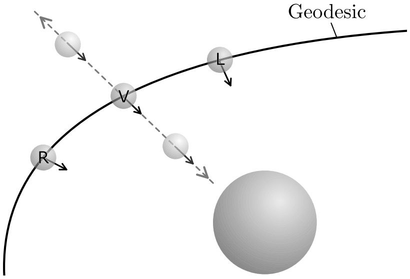


Fig. 1. Camera movement along the line of sight

Analyzing only the camera movement relative to V along the line of sight it can be observed that the new position and V are singular views. This is due to the fact that the camera center of the front camera (we will use the intuitive terms 'forward and backward' as a synonym for 'along the line of sight') lies in the field of view of the rear camera. Accordingly, the images cannot be rectified. The theory described in Section 2 can nevertheless be applied to this case.

The proceeding. Although any camera on the geodesic could be transformed along the line of sight we treat only the case of the first reference camera for simplicity and without loss of generality.

The task is to find a transformation matrix $D_{1\vartheta}$ that describes the camera movement to the front or back where ϑ symbolizes the new camera position. Then we can transform the first reference image according to Equation (6).

We need to determine the homography $H_{\infty 1\vartheta}$ and the epipole $e_{1\vartheta}$. The notation $e_{1\vartheta}$ means that we examine the epipole of the new position ϑ related to the first camera. Accordingly, $H_{\infty 1\vartheta}$ describes the homography between the first camera and the position ϑ. The homography $H_{\infty 1\vartheta}$ is the identity matrix because there is no rotation between the views. An intuitive guess reveals that the epipole $e_{1\vartheta}$ lies in the image center. Thus it is of the form $e_{1\vartheta} = (center_x, center_y, 1)^T$, where $center_x$ and $center_y$ describe the coordinates of the image center. The epipole is scaled to unit length. To control the extent of the displacement the epipole can be rescaled where applicable.

Let s be a scaling factor and $n := \left\|(center_x, center_y, 1)^T\right\|$ the euclidean norm. We get $\widehat{e_{1\vartheta}} = s/n \cdot (center_x, center_y, 1)^T$.

[2] The term zoom is wrong at this point, because we do not change the focal length but move the virtual camera forwards.

Now we can transform points to the virtual view:

$$m_\vartheta \cong H_{\infty 1\vartheta} m_1 + \widehat{e_{1\vartheta}}\gamma_1 = m_1 + \frac{s}{n}(center_x, center_y, 1)^T \gamma_1$$
$$= [I|0] D_{1\vartheta} \begin{pmatrix} m_1 \\ \gamma_1 \end{pmatrix} = [I|0] \begin{bmatrix} 1 & 0 & 0 & \frac{s}{n} \cdot center_x \\ 0 & 1 & 0 & \frac{s}{n} \cdot center_y \\ 0 & 0 & 1 & \frac{s}{n} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} m_1 \\ \gamma_1 \end{pmatrix} . \qquad (7)$$

In order to combine a camera movement $D_{1\nu} = D_{12}^{\nu} (\nu \in \mathbb{R})$ along the geodesic with a movement $D_{1\vartheta}$ to the front into a shared transformation $D_{1\tau}$ we only need to combine the transformation matrices in accordance with Equation (3):

$$D_{1\tau} = \exp(\nu \log(D_{12}) + v \log(D_{1\vartheta})) \text{ with } \nu, v \in \mathbb{R}. \qquad (8)$$

3.2 Camera Movement Orthogonally to the Geodesic and the Line of Sight

We will introduce another virtual camera movement apart from the geodesic. In contradiction to the preceding section, this movement occurs not to the 'front and back' but 'up and down' relative to the original camera pose[3].

The main idea is illustrated in Figure 2. L and R again denote the left and right reference camera, respectively. The virtual camera on the geodesic is labeled with V. The movement of V orthogonally to the geodesic and the line of sight is indicated by a dashed line.

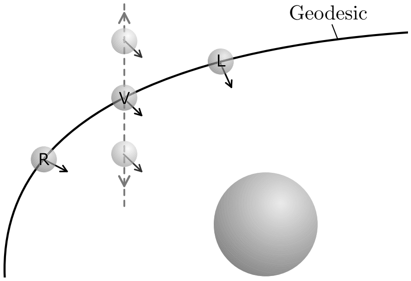


Fig. 2. Camera movement orthogonally to the viewing direction

The proceeding. Without loss of generality only the proceeding for the first reference image is examined. As a result of the fact that the new camera position μ is coplanar to the initial position we have rectified views. Hence, the homography at infinity is the identity matrix $H_{\infty 1\mu} = I$, and the epipoles lie at infinity according to the y-axis: $e_{1\mu} = (0, 1, 0)^T$.

Based on these observations can points in the virtual view μ be calculated with the following formula according to Equation (6):

$$m_\mu \cong H_{\infty 1\mu} m_1 + e_{1\mu}\gamma_1 = m_1 + (0, \gamma_1, 0)^T$$
$$= [I|0] D_{1\mu} \begin{pmatrix} m_1 \\ \gamma_1 \end{pmatrix} = [I|0] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} m_1 \\ \gamma_1 \end{pmatrix} . \qquad (9)$$

[3] Pose: Position and orientation.

Again it is possible to combine these movements with any other displacement with Equation (3). Hence, together with the transformation from the preceding section, we are now able to place the virtual camera anywhere in 3-D space with a camera orientation given implicitly by the orientation of the two reference views.

4 Image Quality Optimization (Utilize Information of Both Reference Images)

In this section a theory will be discussed that allows the usage of the image information from *both* reference images. This is contrary to the processing in [6] where the view syntheses are generated from only one reference image.

Assuming that only the information of the *first* (w.l.o.g.) reference image is used by the procedure in [6], we will now concentrate on including the picture information of the *second* reference image. Thus the transformation for adjusting the second image along the geodesic according to D_{12} is to be found.

Therefore the homography at infinity $H_{\infty 21}$ can be calculated analog to Equation (5). We receive $H_{\infty 21} = H_1^{-1} H_2$ which leads to the coherence $H_{\infty 21} = H_1^{-1} H_2 = (H_2^{-1} H_1)^{-1} = H_{\infty 12}^{-1}$.

The epipoles of the two images are correlated with each other with respect to any homography, i.e. it can be used for example the homography at infinity: $e_1 = H_{\infty 21} e_2$. Using Definition 1 we compute D_{12}^{-1}:

$$D_{12}^{-1} = \begin{bmatrix} H_{\infty 12} & e_2 \\ 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} H_{\infty 12}^{-1} & -H_{\infty 12}^{-1} e_2 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} H_{\infty 21} & -H_{\infty 21} e_2 \\ 0 & 1 \end{bmatrix}$$
$$= \begin{bmatrix} H_{\infty 21} & -e_1 \\ 0 & 1 \end{bmatrix} =: \overline{D_{21}} \qquad \left(\text{Note: } D_{21} \overset{Def.1}{=} \begin{bmatrix} H_{\infty 21} & e_1 \\ 0 & 1 \end{bmatrix} \right) . \tag{10}$$

If the relative affine structure of the second reference image is computed with the inverse homography $H_{\infty 12}^{-1} = H_{\infty 21}$ (equally scaled) as the relative affine structure of the first reference image it is the negative equivalent of the relative affine structure of the first image. Referring to Equation (4) the points of the second image have to be transformed with the version of $\overline{D_{21}}$ adjusted to the negative relative affine structures. This adjusted matrix is D_{21} which is necessary to bring the second camera in the pose of the first camera on the same curve.

A property of the uncalibrated rigid transformation matrix is that the pose of the interstations $t \in \mathbb{R}$ computed for the first image via D_{12}^t and for the second image via $D_{21}^{(1-t)}$ are equal (see [10] together with Equation (10)). The two resulting images are then combined to the final view synthesis, e.g. by averaging them. This is illustrated in Figure 3. The left (L) and right (R) reference view are transformed and yield the *same* virtual view V.

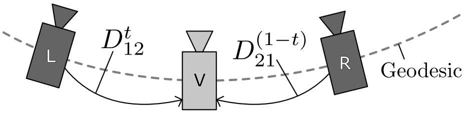


Fig. 3. Combination of two views

5 Experimental Results

In this section we show and discuss some results which were achieved with a MATLAB® implementation of the proposed methods.[4]

Given the two reference images[5] in Figure 4 we computed a smooth video sequence. Some frames of this sequence can be seen in Figure 5. The virtual camera performs a movement from the upper left to the lower right with a simultaneous motion backwards. This can be observed best while following the mound from frame to frame in relation to other objects in the scene. Compared to the results in [6] it can be observed that our 3-D camera placement allows a more flexible path generation. In addition the quality of the virtual images is notably improved as will be discussed in detail now.

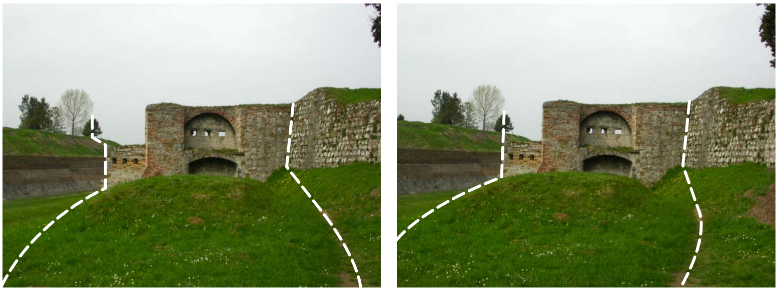


Fig. 4. Pair of stereo images: Left and right reference view. The white dotted lines are added to accentuate the perspective differences.

Figure 6 illustrates the quality improvement achieved by using the image information from both reference images. Two different results of the same scene with a virtual camera describing the extrapolation $D_{21}^{1-(-0,5)} = D_{21}^{1,5}$ are shown. In Figure 6(a) only the image information of one reference image (right camera) is used while the image in Figure 6(b) is received by using the image information of both reference images. It can be clearly seen that the image computed from only one reference image contains much more holes. These holes are parts of the image for which the reference image does not provide any information. The holes still remaining in the virtual images can be closed for example by interpolation.

[4] For videos visit: `http://cvpr.uni-muenster.de/research/viewsynthesis/`

[5] Source: `http://profs.sci.univr.it/~fusiello/demo/synth/`

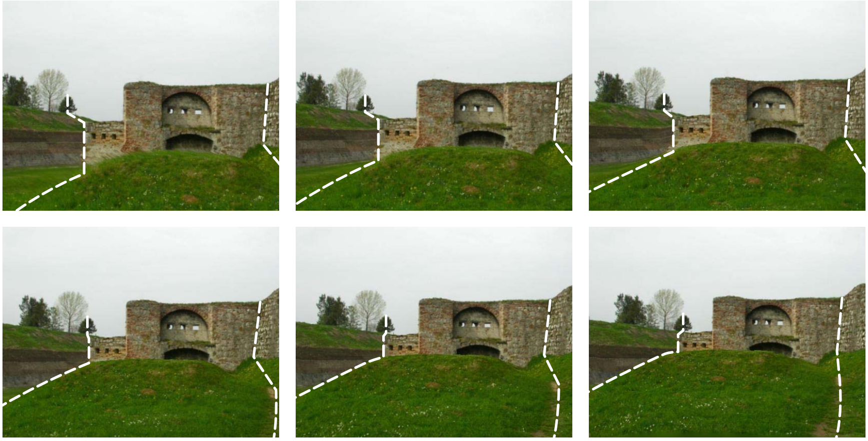
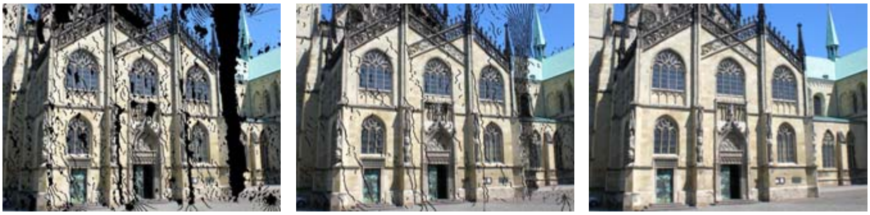


Fig. 5. Some frames of a video sequence computed from the images in Figure 4



(a) One reference image. (b) Both reference images. (c) Interpolation of (b).

Fig. 6. Quality improvement by exploiting the information of both reference images. The erroneous pixels were reduced about 28% by taking both reference images.

Using interpolation on the synthesis resulting from our method shown in Figure 6(b) yields the image in Figure 6(c), which represents a virtual view of excellent quality.

Performance. We evaluated our MATLAB® code on a 64-bit Linux machine with 2.50 GHz and 7.5 GB RAM. According to the three steps of the pipeline, the parameter estimation took about 11 seconds, the stereo analysis about 90 seconds, and the warping varied between 0.8 and 3 seconds per frame for an image size of $640 \times 480 \times 3$ (RGB). The variability in the warping step is due to a changing amount of pixels without image information (holes) which were filled by interpolation.

6 Conclusion

We extended the completely automated view synthesis procedure of A. Fusiello et al. [5,6] to a considerably more flexible procedure with enhanced quality.

After a brief introduction into the basic underlying methods, that allow only for a camera placement on a one-dimensional curve, we developed extensions that provide an arbitrary camera placement in 3-D space. Further we presented an approach to access the image information of all reference images, contrary to [6] where the virtual views are generated based on only one reference image. The gain of image quality could be demonstrated with an example.

Limitations and Future work. We experienced that the image quality of the resulting views almost exclusively depends on the results of the stereo matching procedure. Although the stereo matching step was not subject of this work we predict the most promising advances concerning image quality for view synthesis in general and the presented method in special in the field of stereo matching.

Furthermore, as the view synthesis procedure in [6] can handle more than two reference views, additional reference pictures could be included to provide more picture information for the virtual views. These additional pictures could also overcome the problem of occlusions in non-geodesic synthetic images.

References

1. Seitz, S., Dyer, C.: View Morphing. In: SIGGRAPH 1996, pp. 21–30 (1996)
2. Zitnick, C.L., Kang, S.B., Uyttendaele, M., Winder, S., Szeliski, R.: High-Quality Video View Interpolation Using a Layered Representation. ACM Trans. Graph. 23(3), 600–608 (2004)
3. Criminisi, A., Blake, A., Rother, C., Shotton, J., Torr, P.H.: Efficient Dense Stereo with Occlusions for New View-Synthesis by Four-State Dynamic Programming. Int. J. Comput. Vision 71(1), 89–110 (2007)
4. Avidan, S., Shashua, A.: Novel View Synthesis by Cascading Trilinear Tensors. IEEE Transactions on Visualization and Computer Graphics 4(4), 293–306 (1998)
5. Fusiello, A., Irsara, L.: An Uncalibrated View-Synthesis Pipeline. In: ICIAP 2007: Proceedings of the 14th International Conference on Image Analysis and Processing, pp. 609–614. IEEE Computer Society, Los Alamitos (2007)
6. Fusiello, A.: Specifying Virtual Cameras in Uncalibrated View Synthesis. IEEE Trans. Circuits Syst. Video Techn. 17(5), 604–611 (2007)
7. Shashua, A., Navab, N.: Relative Affine Structure: Theory and Application to 3D Reconstruction from Perspective Views. In: CVPR 1994, pp. 483–489 (1994)
8. Shashua, A., Navab, N.: Relative Affine Structure: Canonical Model for 3D From 2D Geometry and Applications. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(9), 873–883 (1996)
9. Irsara, L., Fusiello, A.: Quasi-Euclidean Uncalibrated Epipolar Rectification. Rapporto di Ricerca RR 43/2006, Dipartimento di Informatica - University of Verona (2006)
10. Alexa, M.: Linear Combination of Transformations. ACM Trans. Graph. 21(3), 380–387 (2002)